Milestone Two Narrative

Noah Jeleniewski

Southern New Hampshire University

A. *Briefly describe the artifact. What is it? When was it created?*

The artifact used for my software design/engineering enhancement is a project I worked on in my *CS 320: Software Testing, Automation, and Quality Assurance* course. The name of this artifact is medicalApplication. The point of this program is to represent a medical system that can add patients/doctors, add medical records, and more. The bigger point of this project in the context of our course was to develop JUnit tests for each relevant function. I originally developed this project several months ago.

*B. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and in software development? How was the artifact improved?*

The reason I selected this artifact for my software design/engineering enhancement is because it had a lot of room for improvement. The components of the artifact that showcase my skills in software development include the ability to perform relevant JUnit tests, discover and fix bugs (some variables were misnamed – passwordAttepts instead of passwordAttempts, for example), my ability to handle exceptions, and my ability to optimize a program. The main problems with the original project was that it was not extensively commented, the main class was cluttered, and the JUnit tests weren’t fully fleshed out. For example, every function had an assertTrue test associated with it, but only some had assertFalse tests, which can be just as important. In this enhancement, I completely fleshed out the JUnit tests, thoroughly commented my code throughout all classes, added several exception handling statements, and fixed the issue of the main class being cluttered. Basically, the main class (App.java) originally contained the logic for prompting the user to enter a password, and if it were correct, then showing the main menu of options. While this is fine, I decided to create a main menu class that is simply referenced in the main class when the program is ran. It just looks cleaner. Overall, this program was improved and optimized greatly with the enhancements I’ve made.

*C. Did you meet the course objectives you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?*

I did meet all of the objectives that I planned to meet with this enhancement in my ePortfolio plan laid out in module one. Specifically, I met the Computer Science program outcomes ‘Develop a security mindset’ (JUnit tests and bug finding/fixing), ‘Demonstrate an ability to use well-founded and innovative techniques’ (Exception handling/optimizing classes), etc. With that said, I may come back to this program to further enhance the exception handling logic, to fully optimize it. As it stands now though, this program looks significantly better than it did before I enhanced it.

*D. Reflect on the process of enhancing and/or modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?*

I think that the biggest thing I learned while enhancing this artifact is that everything can be improved and optimized when looking at a specific program through the mindset of perfecting it rather than just finishing it. While this program originally did accurately perform the functions it was intended to fulfill, there was a ton of room for optimization and improvement, which is what I focused on. I had some challenges regarding figuring out exception handling, and separating a big chunk of the main class into its own class that is simply referenced in the main method. Once I figured these things out, though, the rest of it felt really natural. The JUnit tests were very simple to expand on, and finding bugs like misnamed classes/variables were fun (I enjoy software testing/finding bugs a lot). Overall, enhancing this artifact was a great experience.